

Docket No. A T9-99-655

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: Achtermann et al.

Serial No. 09/438,436

Filed: November 12, 1999

For: Apparatus for Connection
Management and the Method
Therefor§
§
§
§
§
§
§
§

Group Art Unit: 2157

Examiner: Todd, Gregory G.

RECEIVED
CENTRAL FAX CENTER

AUG 31 2005

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450Certificate of Transmission Under 37 C.F.R. 81.8(a)I hereby certify this correspondence is being transmitted via
facsimile to the Commissioner for Patents, P.O. Box 1450,
Alexandria, VA 22313-1450, facsimile number (571) 273-8300
on August 31, 2005.

By:

Michelle Morrow
Michelle MorrowRECEIVED
OIPE/IAP

SEP 01 2005

APPEAL BRIEF (37 C.F.R. 41.37)

This brief is in furtherance of the Notice of Appeal, filed in this case on July 5, 2005.

The fees required under § 41.20(B)(2), and any required petition for extension of time for filing this
brief and fees therefore, are dealt with in the accompanying TRANSMITTAL OF APPEAL
BRIEF.

09/01/2005 FMEK11 00000053 090447 09438436

01 FC:1402 500.00 DA

(Appeal Brief Page 1 of 26)
Achtermann et al. - 09/438,436

REAL PARTY IN INTEREST

The real party in interest in this appeal is the following party: International Business Machines Corporation.

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: Achtermann et al.

Serial No.: 09/438,436

Filed: November 12, 1999

For: Apparatus for Connection
Management and the Method
Therefor

35525

PATENT TRADEMARK OFFICE
CUSTOMER NUMBER

Group Art Unit: 2157

Examiner: Todd, Gregory G.

Attorney Docket No.: AT9-99-655

Certificate of Transmission Under 37 C.F.R. § 1.8(a)I hereby certify this correspondence is being transmitted via facsimile to
the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-
1450, facsimile number (571) 273-8300 on August 31, 2005.

By:

Michelle Morrow
Michelle MorrowTRANSMITTAL DOCUMENTCommissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

ENCLOSED HERewith:

- Appeal Brief (37 C.F.R. 41.37).

A fee of \$500.00 is required for filing an Appeal Brief. Please charge this fee to IBM Corporation Deposit Account No. 09-0447. No additional fees are believed to be necessary. If, however, any additional fees are required, I authorize the Commissioner to charge these fees which may be required to IBM Corporation Deposit Account No. 09-0447. No extension of time is believed to be necessary. If, however, an extension of time is required, the extension is requested, and I authorize the Commissioner to charge any fees for this extension to IBM Corporation Deposit Account No. 09-0447.

Respectfully submitted,

Francois Lamme

Francois Lamme

Registration No. 55,353

Agent for Applicants

Duke W. Yee

Registration No. 34,285

Attorney for Applicants

YEE & ASSOCIATES, P.C.

P.O. Box 802333

Dallas, Texas 75380

(972) 385-8777

Docket No. AT9-99-655

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: Achtermann et al.

Serial No. 09/438,436

Filed: November 12, 1999

For: Apparatus for Connection
Management and the Method
Therefor§
§
§
§
§
§
§
§

Group Art Unit: 2157

Examiner: Todd, Gregory G.

RECEIVED
CENTRAL FAX CENTER

AUG 31 2005

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450Certificate of Transmission Under 37 C.F.R. 81.8(a)I hereby certify this correspondence is being transmitted via
facsimile to the Commissioner for Patents, P.O. Box 1450,
Alexandria, VA 22313-1450, facsimile number (571) 273-8300
on August 31, 2005.

By:

Michele Morrow
Michele Morrow

APPEAL BRIEF (37 C.F.R. 41.37)

This brief is in furtherance of the Notice of Appeal, filed in this case on July 5, 2005.

The fees required under § 41.20(B)(2), and any required petition for extension of time for filing this
brief and fees therefore, are dealt with in the accompanying TRANSMITTAL OF APPEAL
BRIEF.(Appeal Brief Page 1 of 26)
Achtermann et al. - 09/438,436

REAL PARTY IN INTEREST

The real party in interest in this appeal is the following party: International Business Machines Corporation.

**Yee &
Associates, P.C.**

4100 Alpha Road
Suite 1100
Dallas, Texas 75244

Main No. (972) 385-8777
Facsimile (972) 385-7766

**RECEIVED
CENTRAL FAX CENTER**

AUG 31 2005

Facsimile Cover Sheet

To: Commissioner for Patents for Examiner Gregory G. Todd Group Art Unit 2157	Facsimile No.: 571/273-8300
From: Michele Morrow Legal Assistant to Francis Lammes	No. of Pages Including Cover Sheet: 28
Message: Enclosed herewith: <ul style="list-style-type: none">• Transmittal Document; and• Appeal Brief.	
Re: Application No. 09/438,436 Attorney Docket No: AT9-99-655	
Date: Wednesday, August 31, 2005	
Please contact us at (972) 385-8777 if you do not receive all pages indicated above or experience any difficulty in receiving this facsimile.	<i>This Facsimile is intended only for the use of the addressee and, if the addressee is a client or their agent, contains privileged and confidential information. If you are not the intended recipient of this facsimile, you have received this facsimile inadvertently and in error. Any review, dissemination, distribution, or copying is strictly prohibited. If you received this facsimile in error, please notify us by telephone and return the facsimile to us immediately.</i>

**PLEASE CONFIRM RECEIPT OF THIS TRANSMISSION BY
FAXING A CONFIRMATION TO 972-385-7766.**

RELATED APPEALS AND INTERFERENCES

With respect to other appeals or interferences that will directly affect, or be directly affected by, or have a bearing on the Board's decision in the pending appeal, there are no such appeals or interferences.

STATUS OF CLAIMS**A. TOTAL NUMBER OF CLAIMS IN APPLICATION**

Claims in the application are: 1-4, 6-15, 17-26 and 28-33.

B. STATUS OF ALL THE CLAIMS IN THE APPLICATION

1. Claims canceled: 5, 16, and 27.
2. Claims withdrawn from consideration but not canceled: NONE
3. Claims pending: 1-4, 6-15, 17-26 and 28-33.
4. Claims allowed: NONE
5. Claims rejected: 1-4, 6-15, 17-26 and 28-33.
6. Claims objected to: NONE

C. CLAIMS ON APPEAL

The claims on appeal are: 1-4, 8-15, 19-26 and 30-33.

Note: Claims 6, 7, 17, 18, 28, and 29 are rejected under 35 U.S.C. § 112, second paragraph, as having insufficient antecedent basis for the limitations in these claims. Claims 6, 7, 17, 18, 28, and 29 are not being appealed and if the claims under appeal are sustained, claims 6, 7, 17, 18, 28, and 29 will be amended for proper dependency.

STATUS OF AMENDMENTS

There are no amendments after the final rejection.

SUMMARY OF CLAIMED SUBJECT MATTER

Independent claims 1, 12, and 23:

The present invention provides a connection scheduling method. (Specification, page 8, lines 3-12) The present invention determines if a job is available for scheduling. (Specification, page 14, lines 1-14) The present invention determines, in response to said step of determining if said job is available, if a session is available, wherein said session is included in a pool of sessions, said pool of sessions having a preselected one of a set of priority levels corresponding to a priority level of said job and wherein said session effects an execution of said job.

(Specification, page 15, line 1 to page 17, line 12) The present invention launches said session to effect said execution of said job, if said session is available. (Specification, page 17, line 13 to page 19, line 9) The present invention launches an error handling thread in response to an error condition, said error handling thread releasing said session. (Specification, page 20, lines 4-20)

The system recited in claim 12, as well as dependent claims 13-15 and 19-22, may be a bus system comprised of system bus 212; I/O adapter 218; communication adapter 234, memory comprised of read only memory 216 and random access memory 214, and central processing unit 210 performing the steps described in the specification at page 13, line 7, to page 23, line 3, or equivalent. A person having ordinary skill in the art would be able to derive computer instructions on a computer readable medium as recited in claim 23, as well as dependent claims 24-26 and 30-33, given Figure 2 and the corresponding description at page 13, line 7, to page 23, line 3, without undue experimentation.

GROUND OF REJECTION TO BE REVIEWED ON APPEAL

A. GROUND OF REJECTION (Claims 1-3, 12-14, and 23-25)

Claims 1-3, 12-14, and 23-25 are rejected under 35 U.S.C. § 102(e) as being allegedly anticipated by Zolnowsky (U.S. Patent No. 6,779,182 B1).

B. GROUND OF REJECTION (Claims 4, 6-9, 15, 17-20, 26, and 28-31)

Claims 4, 6-9, 15, 17-20, 26, and 28-31 are rejected under 35 U.S.C. § 103(a) as being allegedly unpatentable over Zolnowsky (U.S. Patent No. 6,779,182 B1) in view of Northrup (U.S. Patent No. 6,671,713 B2).

C. GROUND OF REJECTION (Claims 10, 11, 21, 22, 32, and 33)

Claims 10, 11, 21, 22, 32, and 33 are rejected under 35 U.S.C. § 103(a) as being allegedly unpatentable over Zolnowsky (U.S. Patent No. 6,779,182 B1) in view of Northrup (U.S. Patent No. 6,671,713 B2) and further in review of Rangarajan et al. (U.S. Patent No. 6,260,077 B1).

ARGUMENT

A. 35 U.S.C. § 102, Alleged Anticipation, Claims 1-3, 12-14, and 23-25

The Office Action rejects claims 1-3, 12-14, and 23-25 under 35 U.S.C. § 102(e) as being allegedly anticipated by Zolnowsky (U.S. Patent No. 6,779,182 B1). This rejection is respectfully traversed.

As to claims 1, 12, and 23, the Office Action states:

As per Claims 1, 12, and 23, Zolnowsky discloses a connection scheduling method, wherein Zolnowsky discloses:

determining if a job is available for scheduling (job scheduling) (at least col. 5, lines 13-21);

determining, in response to said step of determining if said job is available, if a session is available, wherein said session is included in a pool of sessions (threads), said pool of sessions having a preselected one of a set of priority levels corresponding to a priority level of said job and wherein said session effects an execution of said job (running threads in queue of threads with dispatch priority) (at least col. 6, lines 33-65); and

launching said session to effect said execution of said job, if said session is available (thread (and processor / job) selected for execution) (at least col. 7, lines 17-28; col. 8, lines 43-60).

Office Action dated September 3, 2004, pages 2-3.

Claim 1 is amended to recite subject matter originally presented in claim 5. Claim 1, which is representative of the other rejected independent claims 12 and 23 with regard to similarly recited subject matter, reads as follows:

1. A connection scheduling method comprising the steps of:
 - determining if a job is available for scheduling;
 - determining, in response to said step of determining if said job is available, if a session is available, wherein said session is included in a pool of sessions, said pool of sessions having a preselected one of a set of priority levels corresponding to a priority level of said job and wherein said session effects an execution of said job;
 - launching said session to effect said execution of said job, if said session is available; and
 - launching an error handling thread in response to an error condition, said error handling thread releasing said session.

A prior art reference anticipates the claimed invention under 35 U.S.C. § 102 only if every element of a claimed invention is identically shown in that single reference, arranged as they are in the claims. *In re Bond*, 910 F.2d 831, 832, 15 U.S.P.Q.2d 1566, 1567 (Fed Cir. 1990). All limitations of the claimed invention must be considered when determining patentability. *In re Lowry*, 32 F.3d 1579, 1582, 21 U.S.P.Q.2d 1031, 1034 (Fed Cir. 1994). Anticipation focuses on whether a claim reads on the product or process a prior art reference discloses, not on what the reference broadly teaches. *Kalman v. Kimberly-Clark Corp.*, 713 F.2d 760, 218 U.S.P.Q. 781 (Fed. Cir. 1983). Appellants respectfully submit that Zolnowsky does not teach every element of the claimed invention arranged as they are in the claims. Specifically, Zolnowsky does not teach launching an error handling thread in response to an error condition, said error handling thread releasing said session.

Zolnowsky is directed to a process scheduler or dispatcher for a multiprocessor system for real time applications. The Zolnowsky system uses a dispatcher model that maintains a dispatch queue for each processor and a separate global dispatch queue for unbound higher priority real time threads. Each processor has its own queue and a dispatcher. Each queue has a separate schedule lock associated with it to protect scheduling operations. A processor's dispatcher selects a thread for execution from one of the queues in the system as a candidate thread to execute. When a candidate thread is selected for execution, the processor proceeds to verify against threads in the global real time queue and the processor's own dispatch queue to select a highest priority runnable thread in the system.

Thus, the Zolnowsky system allows the dispatcher to prevent race conditions and minimize lock contention while assuring that high-priority threads are dispatched as quickly as possible. However, there is no section of the Zolnowsky reference that teaches launching an error handling thread in response to an error condition, said error handling thread releasing said session. The Office Action alleges that this feature is taught at column 8, lines 3-17, which reads as follows:

The examination of the priorities on queues involve checking local variable associated with each processor and does not require any locks. For example, priority variable `disp_maxrunpri` can be used to indicate maximum priority level on a queue. Then variable `disp_maxrunpri` can be checked on both the processor dispatch queue and the real time queue using some suitable synchronization algorithm such as Dekker's algorithm to prevent

miscommunication. However, any other suitable synchronization algorithm can be used in alternate embodiments of the present invention. Since the priority variables to be examined are atomic variables that are maintained in each dispatch queue, any scheduling errors caused by selecting a wrong queue will be caught in the a verification step. However, a schedule lock is required to take a thread from a selected queue.

This section of Zolnowsky describes the examination of priorities within the various queues. Zolnowsky teaches using a priority variable to indicate a maximum priority level and making use of a suitable synchronization algorithm to prevent miscommunication between a dispatch queue and a real time queue. Zolnowsky further teaches that any errors between the queues are caught in a verification step. The verification step referred to in this section is clarified by Zolnowsky at column 9, lines 42-50, which reads as follows:

Thus, at step 604, a verification is made as to whether the selected thread is a best possible selection. This requires going back to the high priority real time queue and its own queue, and checking them again to see if there is a higher priority thread newly placed in the high priority real time queue or its own queue. If the selected candidate thread has higher priority than any other thread in either queue, then the processor continues to step 605 to execute the selected candidate thread.

Thus, Zolnowsky teaches a verification of a best possible thread selection by reconfirming the selected threads priority from the other threads in queue. There is nothing in these sections, or any other section of Zolnowsky, that teaches launching an error handling thread in response to an error condition, said error handling thread releasing said session. Even if the verification step were to be considered an error handling thread, the session is not released but merely institutes a thread substitution.

In response to the above arguments, the Examiner in the Final Office Action dated March 9, 2005 states:

... the verification step thread of Zolnowsky (at least col. 8, lines 3-17) is provoked due to errors and resulting in a thread substitution, thus said substitution substituting one thread for another and equivalent to releasing the thread where a schedule lock takes the thread form the selected queue.

As discussed previously, the verification step of Zolnowsky is for determining the best possible thread selection, the verification step is not in response to or provoked by an error, but instead is to prevent any scheduling errors. Zolnowsky does not teach launching an error handling thread

in response to an error condition. Furthermore, if in the verification step, a determination is made that the correct queue has been selected, the process continues uninterrupted. Thus the verification step of Zolnowsky is not in response to an error condition launching an error handling thread.

Thus, Zolnowsky does not teach each and every feature of independent claims 1, 12, and 23 as is required under 35 U.S.C. § 102. At least by virtue of their dependency on independent claims 1, 12, and 23, the specific features of dependent claims 2, 3, 13, 14, 24, and 25 are not taught by Zolnowsky. Accordingly, Appellants respectfully request that the rejection of claims 1, 12, and 23 under 35 U.S.C. § 102 not be sustained.

Furthermore, Zolnowsky does not teach, suggest or give any incentive to make the needed changes to reach the presently claimed invention. Absent the Examiner pointing out some teaching or incentive to implement Zolnowsky such that an error handling thread is launched in response to an error condition and the error handling thread releases the session, one of ordinary skill in the art would not be led to modify Zolnowsky to reach the present invention when the reference is examined as a whole. Absent some teaching, suggestion or incentive to modify Zolnowsky in this manner, the presently claimed invention can be reached only through an improper use of hindsight using the Appellants' disclosure as a template to make the necessary changes to reach the claimed invention.

B. 35 U.S.C. § 103, Alleged Obviousness, Claims 4, 6-9, 15, 17-20, 26, and 28-31

The Office Action rejects claims 4, 6-9, 15, 17-20, 26, and 28-31 under 35 U.S.C. § 103(a) as being allegedly unpatentable over Zolnowsky (U.S. Patent No. 6,779,182 B1) in view of Northrup (U.S. Patent No. 6,671,713 B2). This rejection is respectfully traversed.

Note: Claims 6, 7, 17, 18, 28, and 29 are rejected under 35 U.S.C. § 112, second paragraph, as having insufficient antecedent basis for the limitations in these claims. Claims 6, 7, 17, 18, 28, and 29 are not being appealed and if the claims under appeal are sustained, claims 6, 7, 17, 18, 28, and 29 will be amended for proper dependency.

Claims 4, 8, 9, 15, 19, 20, 26, 30, and 31 are dependent on independent claims 1, 12, and 23 and, thus, these claims distinguish over Zolnowsky for at least the reasons noted above with

regards to claims 1, 12, and 23. Moreover, Northrup does not provide for the deficiencies of Zolnowsky and, thus, any alleged combination of Zolnowsky and Northrup would not be sufficient to reject independent claims 1, 12, and 23 or claims 4, 6-9, 15, 17-20, 26, and 28-31 by virtue of their dependency. That is, Northrup and Zolnowsky, taken either alone or in combination, fail to teach or suggest launching an error handling thread in response to an error condition, said error handling thread releasing said session.

B1. 35 U.S.C. § 103, Alleged Obviousness, Claims 4, 15, and 26

With regard to claims 4, 15, and 26, the Office Action admits that Zolnowsky does not explicitly disclose determining if said connection is an existing connection, and wherein said step of creating said connection is performed if said connection is not an existing connection. However, the Office Action alleges that Northrup teaches this feature at column 4, lines 11-60, which reads as follows:

In the TCS everything is viewed as either being a communication primitive, or, a communication point. The communication primitives are the low-level mechanisms used to provide the physical communication between various processes. The processes participating in the communication are referred to as communication points. Two or more communication points are connected by a communication link using the communication primitives.

The communication primitives are built using the underlying computer operating system intraprocess and interprocess communication facilities and thus are operating-system-specific. On one operating system there may be, for example, five communication primitives supported, while another operating system may support twenty. A communication primitive generally must provide for several operations to be applied such as:

Create: The ability to create an instance of the primitive

Destroy: The ability to destroy an instance of the primitive

Send: The ability to send data to the primitive

Receive: The ability to receive data from the primitive

Cycle: Send a default and receive default messages to/from the primitive

Connect: Primitive specific connection function

Disconnect: Primitive specific disconnection function

Suspend: Primitive specific suspension function

Resume: Primitive specific resumption function

Communication primitives are registered with the Thread Communication Service for the specific operating system the TCS is executing on. The name, the location, and certain characteristics describing the communication primitive are

retained by the TCS for subsequent use. In this context, the communication primitives become a reusable asset, needing to be developed and tested only one time.

Each communication primitive has a shared object, referred to as the communication primitive object, describing the location of the various operations to be applied when using this primitive type. All primitives have the same communication primitive object structure. The TCS will load the communication primitive object at runtime only when requested for use by a communication point.

In a sense, the communication primitive can be thought of as analogous to the physical connection of a telephone on a phone network. A twisted pair telephone would use one primitive while a cellular telephone would use a different primitive.

All this section of Northrup teaches is communication primitives that are the low-level mechanisms used to provide the physical communication between various processes. The processes participating in the communication are referred to as communication points. Two or more communication points are connected by a communication link using the communication primitives. There is nothing in this section, or any other section of Northrup, that teaches or suggests determining if a connection is an existing connection, and, if the connection is not an existing connection, creating a connection.

In response to the above arguments, the Examiner in the Final Office Action dated March 9, 2005 states:

... Northrup teaches connection primitives wherein a thread communication service will run upon request for communication (at least col. 4, lines 11-60). It would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate Northrup's connection primitives into Zolnowsky's system as Northrup teaches communication occurring upon connection commencing. Such as "connection" as stated in the claims is found in Northrup as any connection to another system is going to have an initial connection and for an initial connection to occur it has to be aware that the connection is initial and thus determined to be an initial connection.

Appellants respectfully submit that there is no mechanism in Northrup to determine if a connection is an existing connection. Appellants are not merely claiming initiating a connection, and, while a connection in Northrup may be an initial connection, Northrup does not teach or suggest determining if there is already a connection in existence and then if not initiating a connection.

Moreover, the Office Action may not use the claimed invention as an "instruction manual" or "template" to piece together the teachings of the prior art so that the invention is rendered obvious. *In re Fritch*, 972 F.2d 1260, 23 U.S.P.Q.2d 1780 (Fed. Cir. 1992). Such reliance is an impermissible use of hindsight with the benefit of Appellants' disclosure. *Id.* Therefore, absent some teaching, suggestion, or incentive in the prior art, Zolnowsky and Northrup cannot be properly combined to form the claimed invention. As a result, absent any teaching, suggestion, or incentive from the prior art to make the proposed combination, the presently claimed invention can be reached only through an impermissible use of hindsight with the benefit of Appellants' disclosure a model for the needed changes.

In view of the above, Zolnowsky and Northrup, taken either alone or in combination, fail to teach or suggest the specific features recited in dependent claims 4, 15, and 26. Accordingly, Appellants respectfully request that the rejection of claims 4, 15, and 26 under 35(a) U.S.C. § 103 not be sustained.

B2. 35 U.S.C. § 103, Alleged Obviousness, Claims 8, 19, and 30

With regard to claims 8, 19, and 30, the Office Action alleges that Zolnowsky teaches retrying said steps of determining if a job is available for scheduling, determining if a session is available, and launching said session, in response to an error condition at column 8, lines 11-17, shown above. As discussed above, this section of Zolnowsky describes the verification of the priorities of threads and, if a higher priority thread exist, substituting the higher priority thread.

Moreover, the Office Action may not use the claimed invention as an "instruction manual" or "template" to piece together the teachings of the prior art so that the invention is rendered obvious. *In re Fritch*, 972 F.2d 1260, 23 U.S.P.Q.2d 1780 (Fed. Cir. 1992). Such reliance is an impermissible use of hindsight with the benefit of Appellants' disclosure. *Id.* Therefore, absent some teaching, suggestion, or incentive in the prior art, Zolnowsky and Northrup cannot be properly combined to form the claimed invention. As a result, absent any teaching, suggestion, or incentive from the prior art to make the proposed combination, the presently claimed invention can be reached only through an impermissible use of hindsight with the benefit of Appellants' disclosure a model for the needed changes.

Thus, Zolnowsky and Northrup, taken either alone or in combination, fail to teach or suggest the features recited in claims 8, 19, and 30. Accordingly, Appellants respectfully request that the rejection of claims 8, 19, and 30 under 35(a) U.S.C. § 103 not be sustained.

B3. 35 U.S.C. § 103, Alleged Obviousness, Claims 9, 20, and 31

With regard to claims 9, 20, and 31, the Office Action admits that Zolnowsky does not explicitly disclose retrying is repeated until a predetermined time interval has elapsed. However, the Office Action alleges that the use and advantages for retrying tasks based upon elapsed time is well known to one skilled in the art at the time the invention was made as evidence by the teachings of Northrup at column 10, line 49 to column 11, line 18, which reads as follows:

The Application Process uses the Configuration Administrator Minor Service to administer zero or more components of software from shared libraries. Each component is said to offer a Minor Service. The specifications for the administration of the Minor Services can be provided directly by the Application Service, or, indirectly through a data store monitored by the Configuration Administrator. These specifications can instruct the Configuration Administrator Minor Service to perform the desired operation immediately, at a predefined time (which may be an interval), or, as a result of some event which is later communicated to the Configuration Administrator Minor Service.

The Configuration Administrator Minor Service provides the following operations:

1. Locates specified Minor Services
2. Loads specified Minor Services
3. Executes specified Minor Services
4. Establishes communication channel with the specified Minor Service.
5. Suspends execution of specified Minor Services
6. Resumes execution of specified Minor Services
7. Replaces specified Minor Service with a new Minor Service rerouting communication channels as appropriate
8. Unloads specified Minor Service
9. Provides for manual state retention between replaceable Minor Services
10. Notification

Note that the Configuration Administrator Minor Service operations can be specified to occur at set time intervals; at predefined time periods; as a result of external events; or, as a result of internal events. Events, in this context are registered with the Configuration Administrator Minor Service to denote their occurrence.

While this section of Northrup may teach performing desired operations at a predefined time, this section does not teach or suggest repeating retrying the steps of determining if a job is available for scheduling, determining if a session is available, and launching a session, in response to an error condition, until a predetermined time interval has elapsed. Appellants respectfully submit that it would not be obvious to one skilled in the art at the time the invention was made to retry tasks based upon elapsed time in response to an error condition, as neither of the applied references teach or suggest this feature.

In response to the above arguments, the Examiner in the Final Office Action dated March 9, 2005 states:

... such an event, as described above, being an error state, as it would have been obvious to one of ordinary skill at the time the invention was made that the main reason a service would be retried at a later time could only be due to an "error", such as a thread not being available or an error in the thread itself, and that there would be absolutely no reason to retry something later after it is initially tried except for if there was a problem or "error" condition being met.

Appellants respectfully submit that there is no mechanism in Northrup to perform repeating the retrying steps of determining if a job is available for scheduling, determining if a session is available, and launching a session, in response to an error condition, until a predetermined time interval has elapsed. Additionally, as discussed above, it would not be obvious to one skilled in the art at the time the invention was made to retry tasks based upon elapsed time in response to an error condition, as neither of the applied references teach or suggest this feature.

Moreover, the Office Action may not use the claimed invention as an "instruction manual" or "template" to piece together the teachings of the prior art so that the invention is rendered obvious. *In re Fritch*, 972 F.2d 1260, 23 U.S.P.Q.2d 1780 (Fed. Cir. 1992). Such reliance is an impermissible use of hindsight with the benefit of Appellants' disclosure. *Id.* Therefore, absent some teaching, suggestion, or incentive in the prior art, Zolnowsky and Northrup cannot be properly combined to form the claimed invention. As a result, absent any teaching, suggestion, or incentive from the prior art to make the proposed combination, the presently claimed invention can be reached only through an impermissible use of hindsight with the benefit of Appellants' disclosure a model for the needed changes.

In view of the above, Zolnowsky and Northrup, taken either alone or in combination, fail to teach or suggest the specific features recited in dependent claims 9, 20, and 31. Accordingly,

Appellants respectfully request that the rejection of claims 9, 20, and 31 under 35(a) U.S.C. § 103 not be sustained.

C. 35 U.S.C. § 103, Alleged Obviousness, Claims 10, 11, 21, 22, 32, and 33


The Office Action rejects claims 10, 11, 21, 22, 32, and 33 under 35 U.S.C. § 103(a) as being allegedly unpatentable over Zolnowsky (U.S. Patent No. 6,779,182 B1) in view of Northrup (U.S. Patent No. 6,671,713 B2) and further in review of Rangarajan et al. (U.S. Patent No. 6,260,077 B1). This rejection is respectfully traversed.

Claims 10, 11, 21, 22, 32, and 33 are dependent on independent claims 1, 12, and 23 and, thus, these claims distinguish over Zolnowsky for at least the reasons noted above with regards to claims 1, 12 and 23. Moreover, Northrup and Rangarajan do not provide for the deficiencies of Zolnowsky and, thus, any alleged combination of Zolnowsky, Northrup and Rangarajan would not be sufficient to reject independent claims 1, 12, and 23 or claims 10, 11, 21, 22, 32, and 33 by virtue of their dependency.

In view of the above, Zolnowsky, Northrup and Rangarajan, taken either alone or in combination, fail to teach or suggest the specific features recited in independent claims 1, 12, and 23, from which claims 10, 11, 21, 22, 32, and 33 depend. Accordingly, Appellants respectfully request that the rejection of claims 10, 11, 21, 22, 32, and 33 under 35 U.S.C. § 103 not be sustained.

CONCLUSION

In view of the above, Appellants respectfully submit that claims 1-4, 6-15, 17-26, and 28-33 are allowable over the cited prior art and that the application is in condition for allowance. Accordingly, Appellants respectfully request the Board of Patent Appeals and Interferences to not sustain the rejections set forth in the Final Office Action.


Francis Lammes
Reg. No. 55,353
YEE & ASSOCIATES, P.C.
PO Box 802333
Dallas, TX 75380
(972) 385-8777

CLAIMS APPENDIX

The text of the claims involved in the appeal are:

1. A connection scheduling method comprising the steps of:
determining if a job is available for scheduling;
determining, in response to said step of determining if said job is available, if a session is available, wherein said session is included in a pool of sessions, said pool of sessions having a preselected one of a set of priority levels corresponding to a priority level of said job and wherein said session effects an execution of said job;
launching said session to effect said execution of said job, if said session is available; and
launching an error handling thread in response to an error condition, said error handling thread releasing said session.
2. The method of claim 1 wherein said session comprises a thread.
3. The method of claim 1 further comprising the step of creating a connection to a target system for said execution of said job.
4. The method of claim 3 further comprising the step of determining if said connection is an existing connection, and wherein said step of creating said connection is performed if said connection is not an existing connection.

6. The method of claim 5 further comprising the step of changing value of a job state from a first value to a second value in response to said launching of said error handling thread.

7. The method of claim 6 wherein said first value signals that said job is available for scheduling.

8. The method of claim 1 further comprising the step of retrying said steps of determining if a job is available for scheduling, determining if a session is available, and launching said session, in response to an error condition.

9. The method of claim 8 wherein said step of retrying is repeated until a predetermined time interval has elapsed.

10. The method of claim 9 further comprising the step of registering a callback method in response to an expiry of said predetermined time interval.

11. The method of claim 10 wherein said steps of determining if a job is available for scheduling, determining if a session is available, and launching said session are performed in response to an invoking of said callback method by a target system, said target system for execution of said job.

12. A data processing system for connection scheduling comprising:
circuitry operable for determining if a job is available for scheduling;

circuitry operable for determining, in response to said circuitry operable for determining if said job is available, if a session is available, wherein said session is included in a pool of sessions, said pool of sessions having a preselected one of a set of priority levels corresponding to a priority level of said job and wherein said session effects an execution of said job;

circuitry operable for launching said session to effect said execution of said job, if said session is available; and

circuitry operable for launching an error handling thread in response to an error condition, said error handling thread releasing said session.

13. The system of claim 12 wherein said session comprises a thread.

14. The system of claim 12 further comprising circuitry operable for creating a connection to a target system for said execution of said job.

15. The system of claim 14 further comprising circuitry operable for determining if said connection is an existing connection, and wherein said circuitry operable for creating said connection is operated if said connection is not an existing connection.

17. The system of claim 16 further comprising circuitry operable for changing value of a job state from a first value to a second value in response to said launching of said error handling thread.

18. The system of claim 17 wherein said first value signals that said job is available for scheduling.

19. The system of claim 12 further comprising circuitry operable for retrying said steps of determining if a job is available for scheduling, determining if a session is available, and launching said session, in response to an error condition.

20. The system of claim 19 wherein said circuitry operable for retrying is operated until a predetermined time interval has elapsed.

21. The system of claim 20 further comprising circuitry operable for registering a callback method in response to an expiry of said predetermined time interval.

22. The system of claim 21 wherein said circuitry operable for determining if a job is available for scheduling, determining if a session is available, and launching said session are operated in response to an invoking of said callback method by a target system, said target system for execution of said job.

23. A computer program product embodied in a machine readable storage medium, the program product for job scheduling comprising instructions for:

determining if a job is available for scheduling;

determining, in response to instructions for determining if said job is available, if a session is available, wherein said session is included in a pool of sessions, said pool of sessions.

having a preselected one of a set of priority levels corresponding to a priority level of said job and wherein said session effects an execution of said job;

launching said session to effect said execution of said job, if said session is available; and

launching an error handling thread in response to an error condition, said error handling thread releasing said session.

24. The program product of claim 23 wherein said session comprises a thread.

25. The program product of claim 23 further comprising instructions for creating a connection to a target system for said execution of said job.

26. The program product of claim 25 further comprising instructions for determining if said connection is an existing connection, and wherein said instructions for creating said connection are performed if said connection is not an existing connection.

28. The program product of claim 27 further comprising instructions for changing value of a job state from a first value to a second value in response to said launching of said error handling thread.

29. The program product of claim 28 wherein said first value signals that said job is available for scheduling.

30. The program product of claim 29 further comprising programming for retrying said steps of determining if a job is available for scheduling, determining if a session is available, and launching said session, in response to an error condition.

31. The program product of claim 30 wherein said instructions for retrying are repeated until a predetermined time interval has elapsed.

32. The program product of claim 31 further comprising programming for registering a callback method in response to an expiry of said predetermined time interval.

33. The program product of claim 32 wherein said instructions for determining if a job is available for scheduling, determining if a session is available, and launching said session are executed in response to an invoking of said callback method by a target system, said target system for execution said job.

EVIDENCE APPENDIX

There is no evidence to be presented.

RELATED PROCEEDINGS APPENDIX

There are no related proceedings.